**Working paper – metode**

” Laboratorieeksperiment af typen pretest-posttest, control group-eksperiment”

Indhold

[Projekt 2](#_Toc95388279)

[Uddannelsesinstitution 2](#_Toc95388280)

[Projektledere 2](#_Toc95388281)

[Formål 2](#_Toc95388282)

[Laboratorieeksperiment: Pretest-posttest control group 2](#_Toc95388283)

[Udvælgelse af deltagere 3](#_Toc95388284)

[Population 4](#_Toc95388285)

[Sampling 4](#_Toc95388286)

[Testgruppe, kontrolgruppe og pardannelse 4](#_Toc95388287)

[Antal deltagere 5](#_Toc95388288)

[Ekstern validitet 6](#_Toc95388289)

[Intern validitet 6](#_Toc95388290)

[Metode til selvvurdering 9](#_Toc95388291)

[Selvevaluerings-spørgeskemaer 9](#_Toc95388292)

[Skalavurdering 9](#_Toc95388293)

[Socialt bias 11](#_Toc95388294)

[Anonymitet 12](#_Toc95388295)

[Opsummering 13](#_Toc95388296)

[Bibliografi 13](#_Toc95388297)

# Projekt

Pair programmings indflydelse på vurdering af self-efficacy inden for programmering blandt semitekniske fagprofiler.

# Uddannelsesinstitution

IBA Erhvervsakademi Kolding.

# Projektledere

* Lektor Rikke Lind (rikl@iba.dk)
* Adjunkt Mile Ninkovic

# Formål

Formålet med vores undersøgelse er at undersøge, om semi-tekniske fagprofilers selvvurdering af programmeringskompetencer påvirkes, hvis de arbejder med pair programming (programmering i par) i stedet for individuel programmering.

Den primære researchmetode for vores projekt er eksperimentet. For at kvalificere eksperimentets betydning, supplerer vi det med to spørgeskemaer, hvor deltagerne skal vurdere deres egne kompetencer inden for JavaScript-programmering.

I forbindelse med projektet har vi diskuteret forskellige mulige research-designs til eksperimentet. Nedenfor skitserer vi de overvejelser, vi har haft om vores metodiske valg, og hvad vi har endt med at vælge.

# Laboratorieeksperiment: Pretest-posttest control group

Vores eksperiment er et laboratorieeksperiment af typen pretest-posttest, control group-eksperiment (herefter: ppc-eksperiment), som er et såkaldt sandt eksperiment. (Coleman 2018: 93) Eksperimentet er af typen enkeltfaktor, to-niveaus-eksperiment (two-leveled), hvor faktoren er selvtillid og niveauerne (levels) er hhv. individuel programmering og pair programming (Coleman 2018: 146)[[1]](#footnote-2).

PPC-eksperimentet er kendetegnet ved, at der foretages en observation både før testen (pretest) og efter testen (posttest), samt at der observeres på en kontrolgruppe, som ikke udsættes for testen (control group) (Coleman 2018: 93–94). Det kan beskrives med denne formel, hvor første linje er observationer (O) og stimulus (X) for testgruppen, og anden linje er observationer (O) for kontrolgruppen, samt en streg for at indikere, at de ikke udsættes for stimulus. *R* betyder, at begge grupper er tilfældigt udvalgte (hvilket ikke gælder hele vejen igennem for vores eksperiment, men det vender vi tilbage til):

R O1 X O2

R O3  - O4

I begyndelsen ønskede vi at bygge vores eksperiment op med et ekstra lag, hvor vi før O1 indlagde et ekstra spørgeskema/observation (O0) og en ekstra individuel test/stimulus (X0). Altså:

R O0 X0 O1 X1 O2

R O0 X0 O3  - O4

Det ekstra lag var primært med, fordi vi ønskede at bruge resultaterne af X0 til at vurdere deltagernes faglige niveau – og ud fra dette sætte deltagerne sammen i par, som var nogenlunde egale på dette punkt. Samtidig ville X0 også fungere som en ”træning” af deltagerne til den type opgaver, som testen indebar og dermed – måske – give nogle mere velovervejede besvarelser af spørgeskemaet O1.

Når vi alligevel har valgt at gå væk fra dette design, er det af flere årsager, som kan kompromittere projektets validitet:

1. Jo flere stimuli og observationer, en testdeltager udsættes for, jo større er risikoen for, at deltageren ”regner den ud” og begynder at tilpasse sin opførsel og sine svar efter det, som han tror er projektets formål. Han kan altså udvikle sin egen – evt. ubevidste – dagsorden om enten at bevise eller modbevise det, som han opfatter som undersøgelsens hypotese (Coleman 2018: 91)
2. Den ”træning”, som vi nævnte ovenfor, kan også rent lavpraktisk betyde, at deltagerne oplever det, der kaldes *test, retest gain*: At debliver objektivt dygtigere til/mere sikre i programmering undervejs i eksperimentet, simpelthen fordi de gør det flere gange(Coleman 2018: 91). En evt. stigning i selvsikkerheden i det sidste spørgeskema ville altså være præget af stor usikkerhed om, hvor vidt den skyldtes X0 eller X1
3. Med det ekstra lag ville eksperimentet komme til at strække sig over det, der svarer til en hel skoledag på seks timer. Det ville øge risikoen for maturation confound (Coleman 2018: 133) – at deltagerne pga. varigheden ”ændrer sig” i løbet af afholdelsen, f.eks. pga. sult, kedsomhed eller dalende opmærksomhed – hvilket vi også vil komme ind på senere.

Det vi mister ved ikke at anvende O0 ogX0 er altså primært muligheden for at vurdere deltagernes reelle faglige niveau. I stedet vil vi forlade os på de egenvurderinger, som de udtrykker i spørgeskema O1 og inddele parrene og kontrolgruppen ud fra dette. Det giver naturligvis en usikkerhed – men vi vurderer ikke, at denne er mere betydningsfuld end de ovenfor beskrevne problematikker.

# Udvælgelse af deltagere

Vores eksperiment forudsætter udvælgelse af deltagere i tre tempi: Først udvælgelse af hele deltagergruppen fra populationen, derefter udvælgelse af deltagere til hhv. testgruppe og kontrolgruppe og sidst udvælgelse af deltagere, der skal arbejde sammen via pair programming.

## Population

Vores population er semi-tekniske fagprofiler, som vi definerer som personer med en vis faglig viden og/eller kompetencer inden for med JavaScript-programmering, men som ikke forventes at have specialiseret viden og/eller erfaring inden for hverken JavaScript eller programmering generelt.
Personer med denne fagprofil kan f.eks. være studerende ved semi-tekniske uddannelser som multimediedesigner-uddannelsen, visse cand.it.-studerende eller personer, som engang har stiftet bekendtskab med JavaScript, men som ikke beskæftiger sig med det længere.

## Sampling

Som proxy for populationen har vi valgt at udføre vores eksperiment på en gruppe danske og internationale multimediedesignstuderende fra IBA Kolding Erhvervsakademi fra 3. semester – i alt 78 studerende. Vi har valgt denne gruppe af flere årsager:

1. De multimediedesignstuderende modsvarer den semi-tekniske fagprofil, idét kodning og programmering (herunder JavaScript) udgør ca. 50 % af studiet, mens resten af studiet består af mere ”bløde” fag som design, marketing og forretningsforståelse. De studerende har typisk forskellige interesseområder og kan derfor spænde fra den halvprofessionelle programmør til den ”kodeforskrækkede” designer. Dermed repræsenterer de populationen bredt.
2. Gruppen af studerende har en vis homogenitet i forhold til alder og undervisningsniveau.

Det betyder, at der er mindre risiko for confounds, når vi skal sammenligne resultaterne på tværs. Desuden undgår vi en tidskrævende screeningsproces, og det er nemmere for os at udforme testopgaver, som hverken er for lette eller svære i forhold til deres forventede niveau – selv om dette forventede niveau antages at være meget svingende fra studerende til studerende.

1. Gruppen er – rent lavpraktisk – nem for os at få adgang til.

For at sikre validiteten i eksperimentet bør deltagerne i vores proxy-gruppe udvælges tilfældigt (random sampling), som vi også har været inde på tidligere. I vores tilfælde bliver tilfældigheden varetaget ved, at alle studerende i proxygruppen i forbindelse med undervisningen får tilbud om at deltage i eksperimentet – og som tak for dette vil modtage forplejning samt et gavekort til en biografbillet. Vælger de eksperimentet fra, skal de i stedet arbejde med de samme programmeringsopgaver, uden at vi indsamler data og uden at de får forplejning og gavekort. Det er altså ikke os, der udvælger deltagerne, men deltagerne der vælger, om de vil deltage – med et symbolsk økonomisk incitament. Det giver naturligvis risiko for non-responsive eller self-selection bias, fordi dem, som takker ja til at deltage, muligvis har en særlig profil sammenlignet med dem, der ikke ønsker at deltage (Coleman 2018: 117). Det kan vi dog under ingen omstændigheder undgå, når der er tale om et laboratorieforsøg og ikke f.eks. en observation af dagligdags adfærd – og når vi grundet etik ikke vil tvinge folk til at deltage. Derfor vurderer vi ikke denne bias som noget, vi skal arbejde på at eliminere, men snarere som en omstændighed, vi må have in mente i forbindelse med vores dataanalyse.

## Testgruppe, kontrolgruppe og pardannelse

Vi vil ikke anvende fuldstændig random sampling, når det gælder sammensætningen af par samt fordelingen af deltagere i hhv. testgruppe og kontrolgruppe.

Inddelingen i par bør udføres kontrolleret, så vi sikrer, at parrene så vidt muligt matches på afgørende parametre som fagligt niveau og køn (hvor fagligt niveau som tidligere nævnt vil afgøres ud fra deltagernes egenvurdering i spørgeskema O1). Tidligere studier af pair programming peger nemlig på, at disse parametre har betydning for parrenes kompabilitet (Choi, Deek, and Im 2008: 1117). Således vil vi danne par bestående af to personer med samme køn og nogenlunde samme faglige niveau. Dette er bl.a. for at undgå, at en dygtig programmør ”overtager” arbejdet for en mindre dygtig, eller at deltagerne falder ind i bestemte (køns)roller.

Er det ikke muligt at matche på begge parametre, vil fagligt niveau få førsteprioritet.

Samme studier peger på, at også teamroller målt på Myers-Briggs Type Indicator samt andre personlighedstræk – og sammensætningen af par ud fra disse – spiller en rolle for deltagernes kompabilitet og performance (Choi, Deek, and Im 2008: 1117). For at afgrænse vores undersøgelse og holde et simpelt research-design har vi dog valgt udelukkende at fokusere på fagligt niveau og køn – og vi må så tage forbehold for mulige confounds på personlighed og teamroller ved dataanalysen.

I vores udvælgelse af deltagere til hhv. testgruppe og kontrolgruppe vil vi ligeledes arbejde kontrolleret med en matching-strategi (Coleman 2018: 189–190), så de to grupper kommer til at ligne hinanden mest muligt, og resultaterne derfor også bliver sammenlignelige. For at kontrolgruppen ikke skal indeholde ”dem som det var sværest at parre med andre”, er det dog vigtigt, at vi starter med at opdele i hhv. kontrol- og testgruppe og først derefter danner parrene.

Forholdet mellem deltagere i testgruppe og kontrolgruppe bliver i forholdet 1:2, dvs. dobbelt så mange deltagere i testgruppen som i kontrolgruppen. Vi har valgt denne fordeling, fordi det – pga. par-arbejdet i testgruppen – giver lige mange ”udøvende enheder” i hver gruppe og dermed kan sammenlignes. Hvis der er en deltager for lidt ift. denne fordeling, vil vi lave den fordeling, som kommer nærmest.

## Antal deltagere

Vores eksperiment har størrelse som en kvalitativ undersøgelse, men vi benytter en kvantitativ metode (spørgeskemaet) undervejs. Det betyder også, at eksperimentet ved en senere lejlighed relativt nemt ville kunne skaleres op og aftestes kvantitativt.

Vi har som nævnt en proxygruppe bestående af (potentielt) 78 deltagere. Disse deltagere vil blive inddelt i to grupper, hhv. danske og internationale studerende, og eksperimentet vil køre ad to omgange. Vi deler gruppen, fordi mange af de danske studerende finder det fremmedartet, kunstigt og for nogles vedkommende intimiderende at kommunikere på engelsk, og det måtte derfor forventes at give bias i kommunikationen mellem parterne. Desuden er det nemmere at håndtere en mindre gruppe ad gangen.

Fordi vi henlægger eksperimentet til den normale undervisningstid, og fordi det foregår tidligt på semestret, hvor de studerendes motivation som regel er høj, forventer vi et relativt stort fremmøde på omkring 80-90 % svarende til 27-31 deltagere i den internationale klasse og 35-40 i den danske. Af de fremmødte forventer vi, at langt størsteparten ønsker at deltage i eksperimentet, fordi alternativet (at løse opgaverne alligevel uden at modtage belønning) er mindre tillokkende.

For at bibeholde anonymiteten har vi sat et absolut minimum på 15 deltagere pr. afholdelse. Er der færre end 15, vil vi ikke afholde eksperimentet, da det vil betyde, at vi ikke kan garantere anonymitet i kontrolgruppen, som da vil blive på under 5 individer – samt at vores data bliver mindre valide.

## Ekstern validitet

En udfordring i vores eksperiment er, at pair programming ikke er noget, som vores deltagere – de studerende på multimediedesignuddannelsen – arbejder med på uddannelsen til daglig. Dette er både en fordel og en ulempe for projektets validitet. Det er en fordel, fordi det gør det nemmere for os at afgøre, hvor vidt pair programming reelt har indflydelse på selvvurderingen – uden at vi skal tage højde for, at deltagerne har forskellige erfaringer med pair programming fra tidligere. Det er dog også en ulempe, fordi det sænker undersøgelsens eksterne validitet, altså dens lighed med den virkelighed, som testdeltagerne normalt befinder sig i.

En anden udfordring er de rammer, de studerende skal testes i: Lokaler på deres uddannelsesinstitution. Også her er der både fordele og ulemper. Fordelen er, at det er kendte rammer, hvilket vi antager får deltagerne til at slappe af, modsat hvis der havde været tale om et egentligt ”laboratorium”. Omvendt kan rammerne give reminiscenser til eksamenssituationer og dermed fremme nervøsitet, som man ikke normalt ville finde i programmerings-situation. Dette er i det hele taget en udfordring i vores design, da hele princippet om en test ”lugter lidt” af eksamen og præstationsræs. Her forlader vi os dog på, at netop de studerende, vi vil inddrage i eksperimentet, aldrig har været til en mundtlig eller skriftlig eksamen på institutionen – og at dette kan mindske ligheden med en eksamenssituation.

For at højne den eksterne validitet mest muligt vil vi lade deltagerne anvende egne computere til testen. Simple ting som genvejstaster og musehastighed kan nemlig variere meget fra computer til computer og kan derfor have indflydelse på, hvor nem eller svær, realistisk eller urealistisk en opgave opleves.

## Intern validitet

Coleman beskriver definitionen på et validt eksperiment med tre basale kriterier:

*(1) that the cause comes before the effect, (2) that the cause and effect are related, and (3) that there are no other plausible alternative explanations for the effect.*(Coleman 2018: 130)

Da vores eksperiment udføres som et kontrolleret laboratorieforsøg med (tilstræbt) tilfældig udvælgelse af deltagere og brug af kontrolgruppe, vurderer vi de to første kriterier som opfyldt.

I forhold til det tredje kriterium, som vedrører confounds (andre variable, som har mulig indflydelse på undersøgelsens resultat), er næsten umuligt at undgå i et eksperiment som dette, hvor der indgår mange menneskelige faktorer så som egenvurdering, præstation og samarbejde.

Coleman oplister syv kategorier af confounds: History, Maturation, Testing, Instrumentation, Statistical regression, Selection og Attrition. Nedenfor vil vi gennemgå, hvordan vi ser vores projekt forholde sig til de syv typer confounds.

*History*

History angår hændelser, som optræder før eller under eksperimentet, og som kan påvirke testresultatet. History kan dog nogenlunde kontrolleres i ppc-eksperimenter, hvor kontrolgruppen sikrer, at det ikke er denne hændelse, som er skyld i de eventuelle positive/negative resultater. Vi har desuden diskuteret, hvor vidt vi skulle afholde vores eksperiment på én dag (med den ulempe, at det tager lang tid) eller over flere. Vi har valgt at holde det på én dag netop for at undgå eksterne uens påvirkninger af deltagerne mellem testene.

*Maturation*

Maturation handler om ændringer ved den enkelte testperson som følge af tidens gang, f.eks. stigende alder, livserfaring, sult, kedsomhed, indtrådt sygdom osv. Da vores eksperiment afholdes på én dag, betyder faktorer som stigende alder, øgede erfaringer osv. ikke så meget. Derimod kan kan faktorer som træthed, kedsomhed og sult spille en rolle.

For så vidt som muligt at imødekomme denne confound har vi valgt ikke at gøre testen længere end 1 times varighed – dog 1 ½ time for kontrolgruppen for at kompensere for ”manglende” partner. Vi har desuden – som tidligere beskrevet – valgt at indskrænke det oprindelige research-design, så dagen ikke bliver for lang alt i alt.
Desuden indlægger vi pauser/god tid i eksperimentet og stiller kaffe/te til rådighed for deltagerne. Planen for dagen ser derfor således ud:

|  |  |  |  |
| --- | --- | --- | --- |
| **Tidspunkt** | **Testgruppe** | **Kontrolgruppe** | **Forsøgsledere** |
| 8.20 | Velkomst og tildeling af ID-numre |  |
| 8.30 | Spørgeskema 1 (inkl. intro til udfyldelse) |  |
| 9.00 | Pause | Sammensætning af par/kontrolgruppe ud fra køn, selvvurdering og ID-numre. |
| 9.30-9.45 | Fælles intro til programmeringsopgaver |  |
| Fælles intro til pair programming og regler for programmeringstests |  |
| ”Offentliggørelse” af par og kontrolgruppe |  |
| 9.45 | Lokale 1: Klargøring | Lokale 2: Klargøring |  |
| 9.50 | Test start |  |
| 10.20 | Rollebytning |  |  |
| 10.50 | Spørgeskema 2 |  |  |
| 11.20 | Ekstra ”afledningsopgaver” | Spørgeskema 2 |  |
| 11.40 | Debriefing og tak for hjælpen |  |

*Testing*

Denne confound angår den bias, som ligger i, at folk, der bliver udsat for samme test flere gange, har tendens til at ”score” anderledes anden gang de tager testen – også selvom de ikke har været udsat for nogen stimulus i mellemtiden (Coleman 2018: 134).

I vores projekt indgår to typer af ”tests”: Spørgeskemaerne og programmeringstesten.

I forbindelse med programmeringstesten har vi elimineret testing-confounden, da vi udelukkede den ekstra test X0 fra researchdesignet.

I forhold til spørgeskemaerne må testing-confound’en antages at have en vis betydning. Der er ikke tale om ”tests” i almindelig forstand, eftersom det ikke handler om at præstere eller svare rigtigt på noget. Alligevel kan spørgeskemaet af de studievante deltagere opfattes som en test, f.eks. en test i at ”harmonisere” selvtillid med egentlig kunnen. Dette ser vi dog ikke som et større problem, idet det i så fald netop er affødt af stimulus – programmeringsopgaven – og dermed kan tilskrives denne uafhængige variabel.

*Instrumentation*

Denne confound angår selve målingsmetoden og den bias, der måtte opstå heri, f.eks. når den menneskelige faktor hos os som forskere samt hos deltagerne kan indvirke på udførelsen og tolkningen af eksperimentet (Coleman 2018: 134–135).

Med anonymiseringen af deltagerne i projektet forsøger vi at imødekomme denne problematik. Dog vil anonymiseringen ikke være fuldstændig, idét vi kan se, hvem der er i testgruppe, og hvem der er i kontrolgruppe. Vi vil dog ikke kunne henføre spørgeskemasvarene til konkrete individer, så længe der er minimum 5 deltagere i hver gruppe.
Deltagerne vil også selv vide, om de er i test- eller kontrolgruppe (andet er ikke muligt, når man skal sammenligne effekten af individuel programmering vs. pair programming).

For yderligere anonymisering ville vi skulle indsætte andre facalitatorer end os selv, og testene skulle foregå i helt separate lokaler. I så fald ville vi nærme os princippet for en dobbelt-blindtest, hvor hverken researchere eller deltagere kender fordelingen.
Dette ville dog kræve, at vi kunne råde over 26 lokaler på én gang, hvilket ikke er fysisk muligt på vores lokation. Desuden vil eksperimentet blive spredt over så store fysiske afstande, at vi ville skulle påregne meget længere tid til logistik og løsning af eventuelle tekniske vanskeligheder.

Som et alternativ har vi valgt at blande test- og kontrolgruppe-deltagere i ét rum, således at alle deltagere oplever en grad af snak i rummet. Det giver den udfordring, at testgruppen bliver færdige en halv time tidligere end kontrolgruppen og derfor skal holdes beskæftiget så længe (så de ikke forstyrrer). For ikke at give testgruppen andre forhold end kontrolgruppen er det vigtigt, at de umiddelbart efter opgaveløsningen (og afleveringen af opgaverne) besvarer det sidste spørgeskema. Herefter vil vi lade testgruppen arbejde videre med programmeringsopgaverne for at holde dem beskæftiget, indtil også kontrolgruppen er færdig med at programmere og besvare spørgeskema.

*Statistical regression*

Her har vi at gøre med det dilemma, at ekstreme scores har det med at ”søge mod midten”, når et individ testes flere gange – hvilket er tilfældet med spørgeskemaet. Derfor er det vigtigt, at deltagerne i et eksperiment repræsenterer et bredt fagligt niveau (Coleman 2018: 135).

Dette tilstræber vi ved at understrege i introduktionen, at vi både ønsker deltagere, som er glade for programmering og deltagere, som ikke er. Vi forventer dog, at der vil være en overrepræsentation af programmeringsglade deltagere, da det formentlig er disse, der er mest interesserede i forsøget som sådan.

*Selection*

Vores udfordring på dette punkt er, at vores udvalgte proxy-population kun er på 78 personer, men fordi vi henlægger eksperimentet til undervisningen, forventer vi som tidligere beskrevet en relativt høj deltagelsesprocent.
Se desuden afsnittet om udvælgelse af deltagere.

*Attrition*

Dette vedrører frafald af deltagere undervejs i projektet, og det er bl.a. for at undgå dette, at vi har valgt at gennemføre eksperimentet på en enkelt dag.
Skulle der falde deltagere fra på selve dagen (f.eks. ved pludseligt opstået sygdom), vil det naturligvis påvirke balancen i de to grupper. Vi vurderer dog ikke denne risiko som specielt høj.

# Metode til selvvurdering

## Selvevaluerings-spørgeskemaer

For at afgøre, om eksperimentet har en indvirkning på deltagernes tiltro til deres egne programmeringsevner, vil de blive bedt om at udfylde to spørgeskemaer: Ét før testen og ét efter testen.

Vi anvender spørgeskemaet som metode, selv om der volumenmæssigt er tale om en kvalitativ undersøgelse. Dette har vi valgt, fordi spørgeskemaet med dets skalavurderinger gør det nemt at se udviklingen i selvvurdering for den enkelte deltager.

I spørgeskemaerne tager vi udgangspunkt i Albert Banduras anbefalinger til, hvordan man udformer selvevaluerings-spørgsmål og -skalaer (Urdan and Pajares 2006: 307–337).

## Skalavurdering

Grundlæggende anbefaler Bandura, at man opsætter skalaer med et relativt stort spænd, f.eks. 0-100 og beder respondenten angive et tal for, hvor sikker han er på, at han kan løse en given opgave. Skalaen er suppleret med beskrivelser for minimum-, medium- og maksimum-værdier: *Cannot do at all,* *Moderately certain can do* og *Highly certain can do* (Urdan and Pajares 2006: 312)*,* som løseligt oversat til dansk kan lyde: *Kan helt sikkert ikke*, *Kan måske* og *Kan helt sikkert*. Samtidig påpeger Bandura, at spørgsmålene bør stilles, så respondenten ikke er i tvivl om, at det gælder opgaver, der skulle løses med de forudsætninger, man har her og nu, og ikke opgaver, som respondenten forventer at ville kunne løse engang i fremtiden (Urdan and Pajares 2006: 312–313).

Da programmering er et felt, hvor man som regel benytter hjælpemidler (f.eks. editors og internetfora) og hvor fravær af disse ville have stor indvirkning på deltagernes muligheder for at løse opgaverne, er det vigtigt, at vi i vores spørgsmål gør det helt klart, at disse hjælpemidler er tilgængelige og dermed hvilken præmis, der er for vurderingen.

*JavaScript-programmering*

*Forestil dig, at du sidder ved din computer og skal programmere i JavaScript. Du kan benytte en editor (f.eks. Dreamweaver, NetBeans eller lign.) og har adgang til Internettet, hvor du vil kunne søge efter hjælp og instruktioner, men dog ikke få nogen til at løse opgaven for dig.*

0 10 20 30 40 50 60 70 80 90 100

Kan helt Kan helt
sikkert ikke Kan måske sikkert

 *Nedenfor ser du et antal emner, som knytter sig til programmering i JavaScript.*

*I feltet ”Sikkerhed” skal du skrive et tal mellem 0 og 100, som angiver, hvor sikker du er på, at du ville kunne udføre opgaven, hvis du fik den stillet her og nu.*

*Tallet behøver ikke være rundt.*

*Emnernes rækkefølge er ikke relateret til sværhedsgraden.*

Efter beskrivelsen af settingen vil der blive præsenteret en række JavaScript-opgaver, som deltagerne skal forholde sig til.

Bandura afslutter sin guide med en række eksempler på eller forlæg til selvvurderingsskemaer til brug i forskellige situationer (Urdan and Pajares 2006: 320–335). Skemaerne omhandler meget alsidige emner, bl.a. smertevurdering, bilkørsel, opdragelse, og løsning af akademiske problemer for nu at nævne nogle få.

Nogle af skemaerne beskriver meget specifikke opgaver/udfordringer, som det f.eks. er tilfældet i skemaet *Parental Self-Efficacy*, hvor forældre bedes sætte tal på deres evner til f.eks. at *”Get your children into activities outside of school (e.g. music, art, dance lessons, sports)”* (Urdan and Pajares 2006: 329)*.*

Andre af skemaerne er mere generiske, f.eks. *Problem-Solving Self-Efficacy,* hvor der kun er ét overordnet spørgsmål: *”Please rate how certain you are that you can solve the academic problems at each of the levels described below.”* (Urdan and Pajares 2006: 324)*.* Herefter bedes respondenten tage stilling til, hvor sikker han er på at kunne løse 10% af opgaven, 20% af opgaven, 30% af opgaven osv. Der specificeres ikke, hvilke ”akademiske problemer” der er tale om, men den bestemte form (*”the academic problems”)* antyder, at spørgeskemaet skal besvares i følgeskab med en konkret – og evt. kompleks – forelagt opgave.

Udfordringen med at stille og vurdere komplekse opgaver inden for programmering er imidlertid, at det er meget individuelt, hvornår man vurderer et stykke kode som ”færdigt” – endsige hvornår man vurderer det som 20, 50 eller 80 % færdigt. Programmering er desuden en kreativ disciplin, hvor der som regel er mange måder at løse en given opgave på – og hvor det typisk også er forskelligt, hvor meget man vil mene, der skal til, før noget er færdigt. Et eksempel på dette kan være en mailformular:

1. Er man færdig, når man har kodet selve formularen i html?
2. Er man færdig, når man har kodet formularen i html OG tilføjet css (styling)?
3. Er man færdig, når man har kodet formularen i html og css OG kodet en autosvar-funktion i JavaScript?
4. Er man færdig, når man har kodet formularen i html og css og kodet en autosvar-funktion i JavaScript OG koblet den til en database, hvor de indsendte beskeder lagres og evt. sendes videre?

På grund af ovenstående problematik, som vi vurderer, vil kunne føre til en masse usikkerheder og evt. utilstrækkelighedsfølelse hos deltagerne, har vi valgt at benytte princippet fra forældreskemaet. Deltagerne vil altså blive bedt om at forholde sig, hvor sikre de er på at kunne anvende konkrete delkomponenter inden for JavaScript-programmering, f.eks:

* Variables
* Operators
* Functions
* Arrays
* Osv.

Da spørgeskemaet uddeles i fysisk form – hvilket vi har valgt for at minimere risikoen for tekniske vanskeligheder på eksperiment-dagen – har vi ikke mulighed for at randomisere svarmulighederne. Det giver en mulig bias, eftersom deltagerne måske kan tro, at spørgsmålene stilles i en ordnet rækkefølge, f.eks. ud fra stigende eller faldende sværhedsgrad. Vi anser dog ikke denne bias som voldsomt problematisk, da de udelukkende skal tage stilling til fagbegreber og ikke konkrete opgaver, hvor begreberne sættes i anvendelse, samt da vi i spørgsmålet skriver, at emnernes rækkefølge ikke er relateret til sværhedsgraden.

I spørgeskemaet vil vi desuden spørge til deltagerens nummer-ID og køn – som er nødvendigt ift. parsammensætningen – men ikke alder, da det vil kompromittere anonymiteten unødigt. Før selve JavaScript-spørgsmålet, vil vi indlede med et øve-spørgsmål, så deltagerne kan blive bekendt med skalaen. Dette anbefales af Bandura, da det både kan ”træne” deltagerne i at bruge skalaen, samt i forbindelse med dataanalysen kan afsløre, hvis nogen har misforstået princippet og derfor bør udelukkes fra analysen (Urdan and Pajares 2006: 313). Derudover vil vi bede deltagerne vurdere deres generelle kompetencer inden for JavaScript samt deres oplevelse af at programmere i JavaScript. Dette er for at se, om der sker en generel ændring i hhv. deres egenkompetencevurdering og deres egenoplevelsesvurdering, men vores hypotese er, at dette ikke nødvendigvis vil ændre sig i løbet af undersøgelsen.

De ”generelle” spørgsmål stiller vi til sidst i spørgeskemaet, så deltagerne ikke føler sig ”bondefanget” i det første spørgeskema ved f.eks. først at vurdere deres generelle niveau højt og derefter ”komme til kort” ved de konkrete spørgsmål.

Til sidst vil vi lade der være et tomt felt til åbne kommentarer. Dette er ofte meget værdifuldt i et spørgeskema, da det kan give os input, vi ikke selv havde tænkt på, uddybning af svar samt feedback til det metodiske grundlag, som vi kan tænke med ind i vores dataanalyse.

## Socialt bias

Bandura påpeger, at den sociale kontekst spiller en rolle, når det gælder selvevaluering. Her peger han især på de udfordringer, der opstår, når gruppemedlemmer skal udtrykke deres tillid til gruppens samlede performance-evner (Urdan and Pajares 2006: 316). Dette er ikke umiddelbart et problem i vores undersøgelse, da deltagerne netop ikke skal vurdere deres tillid til gruppens (parrets) samlede evner, men udelukkende deres egne. Vi skal dog være opmærksomme på at være tydelige i vores instruktion heri.

Bandura peger dog også på et andet aspekt:

*Being socially situated, and often interdependently so, individuals’ judgements of their personal efficacy are not detached from the other members’ enabling or impeding activities.* (Urdan and Pajares 2006: 317)

Når deltagerne i vores eksperiment har deltaget i en pair programming-test og efterfølgende skal vurdere deres egne, individuelle evner, kan det være svært (og måske umuligt) at adskille egne evner fra parrets samlede præstation. Dette kan efter vores vurdering både påvirke deres selvvurdering i positiv og negativ retning, afhængig af, hvordan de har oplevet dét at programmere i par.

Vi kan formentlig ikke undgå en usikkerhed i vores resultater pga. denne potentielle bias. Men vi kan tilstræbe at gøre dens indflydelse så lille som muligt ved så vidt som muligt at parre deltagere med samme faglige niveau. Og vi bør igen understrege i spørgeskemaet, at det er tilliden til deres egne kompetencer *her og nu,* de skal vurdere, ikke deres kompetencer i relation til den opgave, de lige har løst.

## Anonymitet

Social bias angår ikke kun deltagernes forhold til de andre deltagere. Det angår også deres forhold til os som undersøgere – især set i lyset af, at vi er undervisere, som i andre sammenhænge skal undervise dem og bedømme deres præstationer på uddannelsen. Der er ikke langt fra denne erkendelse til en forestilling om, at det kan påvirke deltagernes selvvurdering og måske – i en vis udstrækning – transformere den til en selv*fremstilling*. F.eks. kan én deltager vurdere sine egne evner højere end ellers, fordi vedkommende gerne vil gøre et godt indtryk på sine undervisere, mens en anden deltager kan vurdere sig lavere end ellers, fordi vedkommende ikke vil fremstå som utroværdig, hvis det efterfølgende testresultat viser noget andet. Derudover er der risiko for, at nogle deltagere vil sidestille selve testen med en eksamenssituation og dermed blive nervøse og føle, at målet er at præstere godt – hvilket som konsekvens kan påvirke deres præstation og dermed den efterfølgende selvvurdering.

Anonymisering af deltagerne kan hjælpe til at minimere denne sociale bias. Som Bandura da også påpeger: *”Self-efficacy judgments are recorded privately without personal identification to reduce social evaluative concerns.”*(Urdan and Pajares 2006: 314)*.*

Desuden vil anonymitet hjælpe os som undersøgere til ikke at være forudindtagede i vores vurdering af deltagerne, når de skal sættes sammen i par, samt når vi konkluderer på deres selvevalueringer.

I vores eksperiment vil anonymiseringen konkret ske ved, at hver deltager ved eksperimentets start trækker et nummer, som derefter fungerer som deres ID gennem hele eksperimentet. Når vi efter den første test skal danne par til pair programming, kan vi gøre det på basis af testresultaterne alene og ikke på basis af vores forudindtagede holdning til eller erfaring med den enkelte studerende.

ID-nummeret skal deltagerne angive, hver gang de afleverer hhv. et spørgeskema og en programmeringsopgave. På spørgeskemaerne vil det fremgå som et felt, der skal udfyldes. I programmeringsopgaven vil de blive bedt om at aflevere deres opgave på et usb-stik – i en mappe, som er navngivet med deres ID-numre.

En anden, måske mere lavpraktisk måde at sikre anonymitet, kan være at oprette et dokument (f.eks. et Google doc) med numre, som deltagerne så selv markerer som ”taget”. På den måde er der ingen, der har det fulde overblik over, hvilken deltager, der har hvilket nummer. Det giver dog også større risiko for fejl, som senere kan gøre resultaterne ubrugelige (f.eks. hvis to deltagere kommer til at vælge samme nummer).

Anonymiseringsgraden mindskes lidt, når deltagerne efterfølgende arbejder i par, eftersom vi i forbindelse med instruktionen vil se, hvem der deltager i testgruppen, og hvem der deltager i kontrolgruppen.

# Opsummering

For at opsummere vores metodiske tilgang:

* Vi vil udføre to ppc-eksperimenter med op til 78 deltagere (44 + 34), som undervejs opdeles i en testgruppe (2/3) og en kontrolgruppe (1/3). Det absolutte minimum af deltagere er 15 for at sikre anonymitet og validt datagrundlag.
* Deltagerne er multimediedesign-studerende ved IBA og udgør en proxy-gruppe for *semi-tekniske­ fagprofiler*
* Deltagerne skal i løbet af eksperimentet besvare to ens spørgeskemaer og løse et opgavesæt af programmeringsopgaver. Efter første spørgeskema (O1) sætter vi deltagerne sammen i par ud fra fagligt niveau (baseret på selvvurderingen) og køn. Kontrolgruppen dannes ud fra en matching-strategi. Testen udføres herefter i par (testgruppe) og individuelt (kontrolgruppe). Spørgeskema 2 (O2) besvares individuelt.
* Til sidst indsamles spørgeskemaer. Ud fra en komparativ analyse af spørgeskema 1 og 2 og sammenligning mellem testgruppe og kontrolgruppe kan vi se, om pair programming har haft en indflydelse på deltagernes faglige selvtillid.

# Bibliografi

Choi, K.S., Deek, F.P., and Im, I. (2008) ‘Exploring the Underlying Aspects of Pair Programming: The Impact of Personality’. *Information & Software Technology* 50 (11), 1114–1126

Coleman, R. (2018) *Designing Experiments for the Social Sciences: How to Plan, Create, and Execute Research Using Experiments*. SAGE Publications

Urdan, T. and Pajares, F. (2006) *SelfEfficacy Beliefs of Adolescents*. IAP

1. Man kan argumentere for, at vi reelt undersøger på flere levels end to, når vi – som vi kommer ind på senere – ”parrer” vores deltagere ud fra fagligt niveau og køn. Men da vi primært gør dette for at understøtte samarbejdet – ikke for at undersøge indflydelsen på resultatet (det er som nævnt allerede gjort) – vil vi ikke betragte eksperimentet som havende flere end to levels (pair programming og individuel programmering). [↑](#footnote-ref-2)